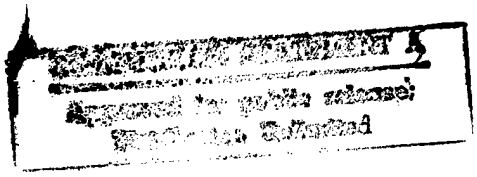
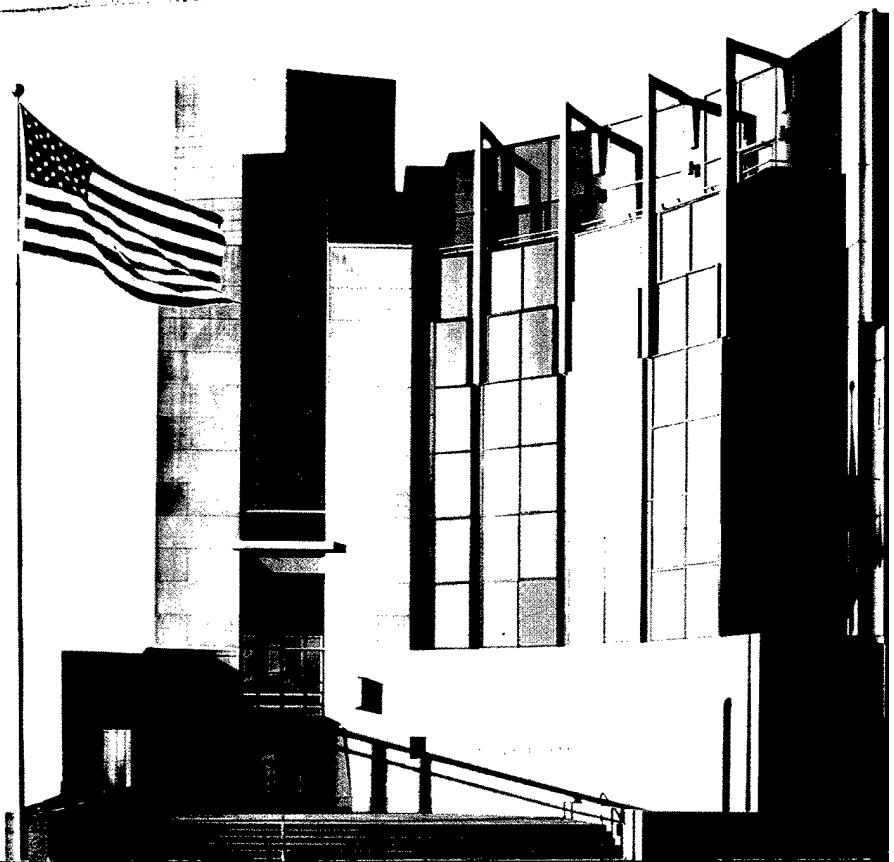
Carnegie Mellon
Software Engineering Institute

Transition Packages for Expediting Technology Adoption: The Prototype Requirements Management Transition Package

Priscilla Fowler
Malcom (Mac) Patrick

September 1998





TECHNICAL REPORT
CMU/SEI-98-TR-004
ESC-TR-98-004

Carnegie Mellon University does not discriminate and Carnegie Mellon University is required not to discriminate in admission, employment, or administration of its programs or activities on the basis of race, color, national origin, sex or handicap in violation of Title VI of the Civil Rights Act of 1964, Title IX of the Educational Amendments of 1972 and Section 504 of the Rehabilitation Act of 1973 or other federal, state, or local laws or executive orders.

In addition, Carnegie Mellon University does not discriminate in admission, employment or administration of its programs on the basis of religion, creed, ancestry, belief, age, veteran status, sexual orientation or in violation of federal, state, or local laws or executive orders. However, in the judgment of the Carnegie Mellon Human Relations Commission, the Department of Defense policy of "Don't ask, don't tell, don't pursue" excludes openly gay, lesbian and bisexual students from receiving ROTC scholarships or serving in the military. Nevertheless, all ROTC classes at Carnegie Mellon University are available to all students.

Inquiries concerning application of these statements should be directed to the Provost, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, telephone (412) 268-6684 or the Vice President for Enrollment, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, telephone (412) 268-2056.

Obtain general information about Carnegie Mellon University by calling (412) 268-2000.

Carnegie Mellon
Software Engineering Institute
Pittsburgh, PA 15213-3890

Transition Packages for Expediting Technology Adoption: The Prototype Requirements Management Transition Package

CMU/SEI-98-TR-004
ESC-TR-98-004

Priscilla Fowler
Malcom (Mac) Patrick

September 1998

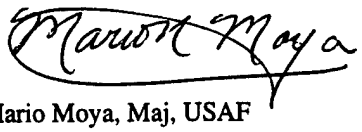
Transition Enabling Program

This report was prepared for the

SEI Joint Program Office
HQ ESC/AXS
5 Eglin Street
Hanscom AFB, MA 01731-2116

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

FOR THE COMMANDER



Mario Moya, Maj, USAF
SEI Joint Program Office

This work is sponsored by the U.S. Department of Defense.

This work was created in the performance of Federal Government Contract Number F19628-95-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 52.227-7013.

This document is available through Asset Source for Software Engineering Technology (ASSET): 1350 Earl L. Core Road; PO Box 3305; Morgantown, West Virginia 26505 / Phone: (304) 284-9000 or toll-free in the U.S. 1-800-547-8306 / FAX: (304) 284-9001 World Wide Web: http://www.asset.com / e-mail: sei@asset.com

Copies of this document are available through the National Technical Information Service (NTIS). For information on ordering, please contact NTIS directly: National Technical Information Service, U.S. Department of Commerce, Springfield, VA 22161. Phone: (703) 487-4600.

This document is also available through the Defense Technical Information Center (DTIC). DTIC provides access to and transfer of scientific and technical information for DoD personnel, DoD contractors and potential contractors, and other U.S. Government agency personnel and their contractors. To obtain a copy, please contact DTIC directly: Defense Technical Information Center / Attn: BRR / 8725 John J. Kingman Road / Suite 0944 / Ft. Belvoir, VA 22060-6218 / Phone: (703) 767-8274 or toll-free in the U.S.: 1-800 225-3842.

Table of Contents

List of Figures

List of Tables

Abstract

This report describes the experience of building and evaluating a prototype transition package for organizations implementing processes in support of the Requirements Management key process area of the Software Engineering Institute's Capability Maturity Model[SM] for Software.[1] This report presents our conclusions based on evaluation and review of the prototype by users typical of the audience targeted for transition packages. Feedback from these users indicated that they were typical "early or late majority" adopters. They found the transition package helpful for orientation and education as part of implementing requirements management practices in their organizations. This report also describes the foundations in research and practice on which the transition package concept is based. We argue in this report that transition packages, as part of a complete "whole product" that includes training and consulting, can be an effective mechanism for expediting the diffusion, adoption, and implementation of important technologies. Finally, we describe what we now know about creating transition packages and how they might be used.

[SM] Capability Maturity Model is a service mark of Carnegie Mellon University.

[1] A preliminary version of this technical report appeared in [Fowler 98].

Acknowledgments

Many people have participated with us in exploring the transition package as a means of accelerating technology adoption. We thank the following people who helped as reviewers, as contributors of ideas, time, and materials, and as supporters of the transition package concept:

Anonymous reviewers and users of the Web site

Reviewers of drafts of this report: Lana Cagle, Eileen Forrester, Frank Hollenbach, Linda Levine, Eamonn McGuinness, Nancy Mead, Brian Middlecoat, Mike Phillips, Alan Prochaska, Jock Rader, and Dave Zubrow

Participants in the 11/96 and 3/97 workshops: Rick Barbour, Lana Cagle, Regina Cohen, Gilles des Rochettes, Donna Dunaway, Frank Gangemi, SuZ Garcia, Craig Hollenbach, Brian Larman, Linda Levine, Michelle Loo, Janice Marchok, Linda Fay McCalla, Eamonn McGuinness, Nancy Mead, Charles Myers, Mike Phillips, Jock Rader, Brian Snarzyk, Cecilio Valdez, and Dao Vu

Our co-authors for the precursor conference paper for the 1998 International Conference On Requirements Engineering: Anita Carleton and Barbara Merrin

Editor and advisor: Sandy Shrum

Web site developer: Melissa Kacik

Dick Phillips, for coining the phrase "transition package"

Mike Phillips, our manager, John Goodenough, the SEI Chief Technical Officer, and Steve Cross, Director of the SEI, for supporting the work

We particularly thank the organizations that donated materials to the Prototype RM Transition Package Website: Amoco, Litton/PRC, Naval Air Systems Command, Naval Oceanographic Office, Office of Management and Budget, Sacramento Air Logistics Center/McClellan AFB, Synertech, and the USAF Material Systems Group.

1 Executive Overview

Around the world, the software and information technology (IT) development communities are challenged by growing demands in industry and government to produce higher quality software, and to do it faster and more predictably. To meet this challenge, software and IT professionals must become far more adept at adopting and implementing new technologies and practices in their organizations [Goldenson 95, Klein 95, Klein 96, Leonard-Barton 92]. One possible solution to this demand for highly efficient and predictable adoption is the transition package (TxP). A transition package is a kit-based approach to providing the materials needed to use new technologies and practices as well as to introduce technologies and practices into organizations. Transition packages may help software and IT professionals apply, within their own organizations, the principles developed for packaging and marketing commercial software. Using these principles can expedite the adoption and implementation of maturing technologies and practices.

Work at the Software Engineering Institute (SEI) has included a series of projects focused on developing a systematic and reliable approach to technology introduction that would expedite organizational implementation of new technologies. One such approach, the Technology Transfer Model (TxM), was co-developed with Xerox Corporation. The TxM was generally well received but criticized for a lack of accompanying examples. Users of the prototype TxM at Xerox and later at Union Switch & Signal felt that technology-specific versions of the process model should be developed, each accompanied by technology-specific examples, templates, and related materials [McAndrews 1997]. In the absence of these materials, teams using the model were forced to spend time finding or creating examples and had to use trial and error to refine what they found to fit their organization's needs. This experience, combined with a re-interpretation of Moore's "whole product" concept for application within organizations (versus in the market place) led us to the idea of a transition package [Moore 91].

The transition package concept was explored in a workshop for people who had already implemented the Requirements Management (RM) practices described in the Software CMM® or were in the process of doing so [Fowler 97a, Fowler 97b]. We wanted to learn what they thought about the idea of pre-packaging materials to help with implementation. Participants in this workshop shared their approaches to implementing requirements management, and—most importantly for us—provided enthusiasm, example materials, and direction for evaluating the transition package idea.

® CMM is registered in the U.S. Patent and Trademark Office.

We then developed a prototype of a transition package for the RM key process area (KPA). We used materials from organizations that had implemented RM. Organizations donating artifacts for use in the prototype included Amoco, Litton/PRC, Naval Air Systems Command, Naval Oceanographic Office, Office of Management and Budget, Sacramento Air Logistics Center/McClellan AFB, Synertech, and the USAF Material Systems Group. The SEI also provided materials from its training course on requirements engineering [Zelesnik 92]. Of about 100 artifacts donated, fifty-nine artifacts, ranging from policy examples and templates to example plans from actual change teams implementing RM, were included in the prototype package. The package was developed as a password-protected World Wide Web site with three "views" into the collection of artifacts:

- a Software CMM view based on the common features of the RM key process area
- a view based on the eight activities of the technology transfer model
- a view based on an index organized by artifact type (examples, templates, guidance, and checklists)

After developing the prototype, we organized its trial use and evaluation. We recruited participants from organizations that were currently implementing RM and looking for help—those typical of the audience for whom the transition package was designed—to participate in trial use of the package and provide feedback. The prototype was available to trial participants from late July 1997 until the end of October 1997.

Participants completed a pre-trial survey, giving information about their organizations, the type of software they developed, and their efforts to implement RM. At the end of the trial period, we interviewed participants to learn how they used the materials and whether the prototype had been useful to them. All of the participants interviewed said that they benefited from using the materials. After four months, we talked to the trial participants again. Several had discontinued their improvement efforts, several had found other sources of materials, and a few continued to use the prototype transition package materials and found them beneficial. In both sets of feedback, participants felt that transition packages should be built for the KPAs of the Software CMM by the SEI. They attributed their incomplete use of the prototype RM transition package to factors outside the package, including changes in organizational direction, redirection of change team efforts, changes in job assignments, and changes in management. According to the Software CMM, these factors represent many of the risks for projects in a typical Level 1 organization.

To generalize our experience from producing and evaluating the prototype RM transition package and before recommending how best to create other transition packages, we revisited software engineering technology transfer research and practice reports identified earlier. We confirmed that the diffusion and adoption literature concerned with the implementation of software and Information Technology applies to the development of transition packages. A key example of this literature is Moore's *whole product* concept mentioned above [Moore 91]. Change teams understood this concept because their members generally had experience with purchased, packaged software and could see how internal implementation of technology is analogous to the implementation of packaged software. These teams appreciated how a starter set or kit-based approach to implementation, represented by a transition package, might help them implement technology-based change.

As a result of our research, experience, and examination of theory, we believe transition packages are an important part of a support mechanism for introducing new technologies. Particularly for *early majority*[2] and later adopters, the availability of these materials combined with other "whole product" services and products may be a prerequisite for successful adoption. Those technologies introduced without transition packages are less likely to find acceptance after experiencing initial success among the *innovator* and *early adopter* groups, who are able to build their own transition-package equivalents. We believe any research and development effort that seeks to bring a technology into widespread use as a new technology "standard" will benefit from developing transition packages as a complement to more traditional marketing, sales, and support from the technology "push" side. Appendix A summarizes our understanding of how to build transition packages.

[2] As described in both Everett Rogers' *Diffusion of Innovations* and Geoffrey Moore's *Crossing the Chasm*, adopter populations can be characterized in several groups depending upon when they begin to use a technology new to them [Rogers 95, Moore 91]. The *innovators* are the technology enthusiasts who first try out a new technology. *Early adopters* are those who come next, taking chances with technologies that the *innovators* have endorsed, with the goal of solving some pressing, often competitive, need. The *early majority* are the pragmatists who adopt a new technology when it has been demonstrated to be useful to others in their domain. The *late majority* change to the new technology when it becomes a standard that they must support or risk being left behind. The *laggards* are those who will not use the new technology at all. Each of these groups has different motivations, needs, and goals. Innovators and early adopters are the only groups willing to take a "do it yourself" approach. The later adopters want turn-key systems and approaches. To be successful, anyone attempting to introduce new practices into an organization must recognize and accommodate these differences.

2 Transition Packages for the Software Engineering Community

To improve the practice of software engineering, the SEI and others with similar missions must change the behavior of hundreds of thousands of technical professionals working on software products and software-intensive systems. This change requires technology transfer and diffusion of innovation on a grand scale [Fichman 92, Kwon 87, Rogers 95, Tornatzky 90]. And indeed, this is occurring continuously. A vast network of universities educates individuals in both emerging and established technology areas. Information about technology is propagated through mass media such as the Internet and the World Wide Web. Commercial enterprise delivers new technologies through products and services.

2.1 The Need for Transition Packages

Actual adoption into practice of new technology-based solutions and products is slow compared to the speed with which solutions are proposed and developed, and with which information about solutions and products is disseminated. Recent graduates bring new ideas and approaches to organizations but are seldom the most influential employees. Knowledge of how to apply technology and products is not always available, and organizations cannot adopt every new technology or product that looks attractive. Thus, the technology selection and adoption process in organizations becomes a bottleneck in the diffusion and use of software engineering technologies.

Investigations into the nature of technology maturation, diffusion, adoption, and implementation have clarified the causes of slow rates of diffusion and use. A consistent, albeit implicit, finding of these investigations is that the same techniques that expedite adoption, diffusion, and implementation of commercial products can be used for technologies for software engineers [Levine 94, Morton 83]. In particular, an approach that holds promise is packaging the process and materials of technology *introduction* with example processes and materials for the *use* of a technology; for example, *The Software Inspections Process* [Strauss 94].

Thus, improvement in the state of the practice in software engineering depends upon improved methods of technology introduction. These methods can limit or enable the adoption of useful new software engineering technologies and practices [Morton 83, Przybylinski 87, Orlikowski 93, Levine 94, Fowler 94]. In his groundbreaking book, *Crossing the Chasm*, Geoffrey Moore rationalizes success on the *push* side of the technology transfer equation—that is, how a marketer disseminates products more widely and

effectively by means of whole products [Moore 91]. We have adapted Moore's concepts for use on the *pull* side of technology transfer, that is, where change agents get technology (including products) adopted and used in practice within their organizations. We call this approach a *transition package* [Fowler 97a, Fowler 97b, Fowler 98]. This report describes our work to understand the application of the transition package concept.

2.2 The Transition Package Concept

As we envision it, a transition package for any software technology—be it Fagan software inspections, object-oriented design, or a Software CMM KPA such as RM—is a designed and integrated set of components for use in the introduction and application of that technology [Fagan 76, Jacobson 92, Paulk 94]. Ideally, a documented process of introduction (including customization guidance) enhances this set of components for use at the project level, and it includes a deployment strategy for rolling out the technology across multiple projects or an entire organization. A transition package contains examples, templates, checklists, and guidance in a particular technology area—all of the materials that a team responsible for facilitating technology-based change would need to get their organization started with new practices. (See Figure 1 for an example of a whole product for adopting software inspections based on a description of the AT&T Bell Labs software inspections program [Ackerman 83].)

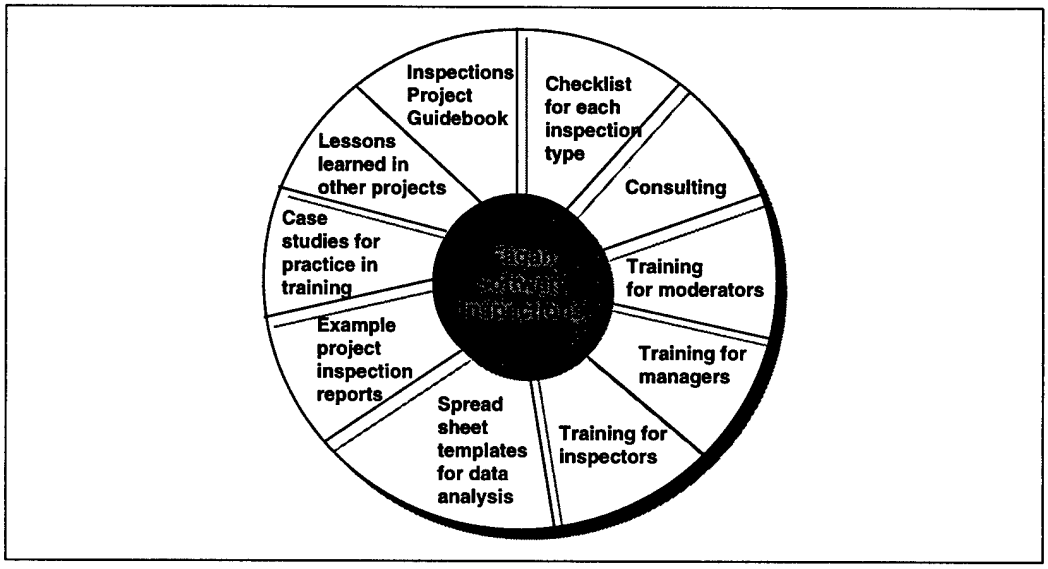


Figure 1: A Whole Product Example Based on Fagan Software Inspections as Implemented at AT&T Bell Labs

As noted earlier, innovators and early adopters seem inclined to solve adoption problems by developing their own artifacts, examples, and guides. In contrast, early and late majority adopters look for standard materials that can be tailored easily to suit their transition situation and needs. Many organizations acquire these materials from consultants, through membership in user groups, or by partnering with other organizations attempting to solve similar problems. Sometimes books, such as *Software Metrics: A Company-Wide Approach*

that covers implementing software metrics, or *Software Inspection Process* that covers introducing software inspections, constitute a transition package of sorts [Grady 87] [Strauss 94]. These books are collections of materials developed during experimentation by early adopters of these technologies, who then packaged the materials for the use of later adopters. Some large organizations build their own transition packages to expedite software engineering technology introduction and implementation [Culver-Lozo 95, Hollenbach 96, Rader 96].

To bring easier technology introduction to the bulk of organizations that are candidates to adopt and implement a technology, transition packages address a key part of the whole product necessary to implement a new technology—the part that provides example materials and guidance for adapting these materials for use. Majority populations prefer, and are better able to adopt, technologies that are mature, packaged, and predictable in installation and use.

In general, we believe that users (or potential users) of transition packages are people who are responsible for identifying and coordinating activities related to technology introduction in their organization. These change agents in software organizations typically include the following people:

- software engineering process group (SEPG) members
- process action team / technical working group members
- advanced technology group members

Other potential users and builders of transition packages, include

- champions, who are advocates with influence—they may be management sponsors or other suppliers of resources for building transition packages
- legitimizers, who are domain experts that influence champions
- practitioner experts in domain-specific technology transfer

Individuals from these groups who can see the strategic uses of a transition package may sponsor or perform its development, or may insist that change agents find existing transition packages to use as the basis for technology-based solutions to problems.

3 The Prototype Transition Package

In this section we describe how the transition package prototype was developed, tested, and evaluated. We begin by discussing why the requirements management key process area from the Software CMM was chosen for a first trial of the transition package concept.

3.1 Why Requirements Management?

We selected requirements management as the first technology to use for evaluating the concept of a transition package because RM is a common problem area and is addressed as a key process area in the Software CMM. These characteristics meant that a large number of organizations had already grappled with issues of RM and we could tap the lessons they had learned. In addition, we chose RM because we had recent experience working with two organizations that introduced RM practices. Thus, we were familiar both with the RM content, and with issues related to introducing RM. Also, from a technology implementation perspective, the RM key process area, with only three activities (i.e., requirements are documented, changes are reviewed, and changes are reflected in plans and work products), appeared to be simpler than other key process areas.[3]

For our prototype package, we wanted a technology that was being adopted by users who could be characterized as part of a *majority* adopter population—that is, the 70% or so of organizations that are candidates to adopt a particular technology after the innovators and early adopters have demonstrated its worth.

[3] The initial assessment based on the transition literature indicated that fewer activities to implement meant it might take less time to get new practices into place for RM than it might for other KPAs with more activities. [Leonard-Barton 92]. However, RM is at the intersection of the software project and the rest of the organization, and can involve individuals from a range of non-software functions. Because of this central role, implementing RM can involve negotiating with people who don't have a software background, who may not understand the importance of RM, and over whom software people seldom have direct control. These factors make RM a more difficult process area to implement than some others (e.g., Configuration Management) that are entirely under the control of the software engineers. However, RM does have the advantage of being a carefully bounded subset of the very large field of requirements engineering [Davis 93].

The SEI-developed CMM for Software has been in use since 1987 (the present version since 1991) and hundreds of organizations now use the CMM as a reference model for software process improvement[4] [Hayes 95, Paulk 94]. We assumed that organizations that are currently implementing the key practices at the initial level were starting to use the CMM some years later than the innovator and early-adopter organizations that began using it when it was first released. Thus, these organizations were likely to belong to the early majority population of Software CMM adopters.

To summarize, for our prototype transition package, we wanted a technology that was

- familiar to transition package project team members
- fairly simple to implement
- widely used and somewhat mature

Requirements management seemed to fit all of these criteria.

3.2 The Software CMM and RM

The Software CMM, first proposed in a draft version in 1987, is a five-level model of software process management maturity, organized to describe management capabilities clustered into key process areas [Paulk 94]. KPAs provide helpful "chunking" of technology that can be the basis for planning software engineering technology adoption in organizations [Fichman 92].

The KPAs describe maturity capabilities ranging from project-level engineering management capability (at Software CMM level 2) through statistically-driven process improvement capability (at Software CMM level 5). Level 1 is where most organizations are presumed to start and is the ad-hoc state of management "incapability." Details of this model are described in a range of sources [Paulk 94, Dymond 95].

The KPAs at Software CMM level 2 support projects within the organization, and address

- the management of customer requirements for software
- the planning of the projects based on those requirements
- the tracking of projects' progress against plans
- the selection and management of subcontractors contributing to project work

[4] In a 1996 presentation at the SEI, Bill Peterson, director of the SEI's Process Program, reported that in 1989, 46 people attended the first SEPG National Meeting; in 1995, 1,248 attended the (renamed) SEPG Conference. He also reported that there are now 54 Software Process Improvement Network (SPIN) groups, representing a regional or national group of SEPGs. [Authors' note: There are now 89 SPIN groups, and attendance at the SEPG conference has been 1400-1500 each year from 1996 through 1998].

- the planning and execution of quality assurance activities for the project and for the work products
- the maintenance of integrity for the products to be delivered by the project

In the Software Capability Maturity Model, when these six KPAs are in operation in all projects within an organization, the work the organization does is *repeatable* with some degree of confidence.

The purpose of the requirements management KPA is to describe what projects in a software organization must do to achieve these two goals:

- Goal 1: System requirements allocated to software are controlled to establish a baseline for software engineering and management use.
- Goal 2: Software plans, products, and activities are kept consistent with the system requirements allocated to software [Paulk 94].

The RM KPA is at the front end of software development, where the customer is involved, and it is often the first KPA chosen for improvement by organizations using the Software CMM as a "road map" for process improvement [McFeeley 96]. Organizations find that establishing requirements management discipline and capability prior to implementing corresponding practices in project planning, for example, is much easier than attempting to put project planning in place without having a firm basis in RM on which to build. However, because RM is usually among the first areas selected for improvement, its easy implementation is more doubtful than the implementation of later KPAs.

3.3 Developing the Prototype

Based on a series of workshops, we developed requirements for the RM TxP from experts and from potential users, acquired the contents, and designed the TxP for delivery via a secure Web site.

3.3.1 Determining Contents

With an analysis of RM and its potential for use in a prototype transition package in mind, a by-invitation workshop was designed to draw together people from organizations that had implemented or were in the process of implementing the RM KPA [Fowler 97b].

The workshop, held in November 1996, convened participants from nine organizations including the SEI. Participants provided a set of strong recommendations for what should be included in an RM transition package. In addition, they endorsed the idea that the SEI could and should develop transition packages, at least for the Software CMM key process areas (including requirements management) and possibly for other SEI-supported technology programs.

The workshop participants listed 136 candidate artifacts and proposed categories for organizing them within an RM transition package. After evaluating the results of the workshop, we proposed and then began the steps to develop a prototype RM transition package.

The prototype development steps were to

- assemble components for a prototype transition package for requirements management
- provide the prototype to organizations that were implementing requirements management
- learn from these organizations whether the transition package made a difference in their RM work
- test our assumptions about
  - transition package users and how they would use the package
  - the content of the package
  - how to fund development
  - the distribution of transition packages

Our goal was to learn what a transition package as a product should be, based on experience with the prototype.

The workshop had produced a complex set of suggestions for what we had hoped would be a relatively straightforward product. There was no time in the workshop for participants to write descriptions of each artifact they suggested, or to agree on a good way to organize and present the artifacts in a transition package. It was clear that more work needed to be done to simplify the requirements for the prototype package before we could determine its specifications. At the SEPG Conference in March 1997, we invited those who participated in the workshop, as well as others who had shown interest, to meet. We asked them to create descriptions of the artifacts identified by participants in the earlier workshop, and to reorganize the categories and artifact names. We also used the SEPG conference to host birds-of-a-feather sessions to describe the prototype development effort, to invite broader participation, and to get more ideas for the prototype from conference attendees.

Using this information, we derived the specifications and the categories for the contents of a prototype transition package for RM. We then began to look for sources of artifacts to populate the package. Feedback had indicated that it would be best if these artifacts represented the work of people in organizations with experience implementing RM. With this in mind, we first contacted those who had attended our workshops or the birds-of-a-feather sessions. Many people responded with donations of candidate materials for the prototype TxP, ranging from meeting minutes and technical notes on RM to process descriptions and policy examples. In addition, we found example materials in an SEI course on systems engineering, including checklists and a template for requirements specifications [Zelesnik 92]. We asked SEI staff who had served as software process improvement consultants to request artifacts from their customers. We found good example artifacts on the Web and asked permission of their owners for use of these.

We were able to gather nearly 100 artifacts of which we used 59; we obtained permission from the donors to allow us and the transition package prototype users to read, copy, and use them without restriction during the prototype period. To obtain this permission, we agreed to protect donor anonymity by removing organization names and identifiers from the artifacts and, in some cases, by changing industry references. Otherwise, we used the artifacts intact to save resources, and because experience was absent about what would add value to the collection or to individual artifacts. Thus, while there was a rich variety of materials, the collection of artifacts in the prototype was not integrated and did not have a consistent look and feel. We did not investigate whether this "roughness" inhibited use, was unimportant to users, or gave the collection character.

The materials gathered reflected a range of organizational experience in introducing RM. Some donors were just beginning to work toward level 2; most contributing organizations were already at Software CMM level 2; two were at level 3. Table 1 shows the artifacts that were included in the RM TxP, organized by the first three artifact types (Checklists, Examples, and Templates). Table 2 shows the artifacts in the fourth category, Guidebooks/Guidance. This last category is the largest because it contained several sets of related artifacts; these represented documented RM processes in two organizations and RM processes excerpted from the Software Process Framework [Olson 94].

Checklists:
Training requirements checklist (from SEI course)
SRS contents list (from SEI course)
Examples:
RM "as is" report (current state of RM practice in one organization)
RM Technical Working Group Charter
RM Technical Working Group Tactical Action Plan
Meeting minutes from selected RM Technical Working Group meetings
List of RM Artifacts from a RM Technical Working Group
RM "to be" process charts (desired state of RM practice in one organization) (ETVX format)
Gap analysis of "as is" RM process, compared to CMM RM KPA, from a RM Technical Working Group
Project schedule from an RM Technical Working Group
Multi-level partial strawman RM process flow (incomplete draft of flow)
"ABC" company's RM process flow (corporate view, incorporating tool use at division and project levels)
RM Policy example #1 (from SEI course)
RM Policy example #2 (from SEI course)
Example RM policy (organization level)
Templates:
Template for project level RM policy
One organization's Software Requirements Specification (SRS) Template
Data Item Description for SRS (DID-SRS)
Data Item Description for Interface Requirements Specification (DID-IRS)
Data Item Description for Consolidated Software Requirements Document (DID-CSRD)
Project Requirements Form

Table 1: Artifacts in the Checklist, Example, and Template Categories

Guidebooks/Guidance
Requirements Management Technical Note
List of RM Artifacts from an RM Technical Working Group
How RM relates to all the CMMs
Material from one organization's RM training course
List of RM artifacts from an RM Process Action Team (PAT)
Requirements specification guidance from one organization
RM Guidebook from Naval Air Systems Command
Procedure RM-P-1: Establish a Functional Review Board
Procedure RM-P-2: Perform the FRB Function
Procedure RM-P-3: Review Emergency Change Request
Procedure RM-P-4: Allocate System Requirements
Procedure RM-P-5: Derive and Analyze Software Requirements
Procedure RM-P-6: Trace Software Requirements
Procedure RM-P-7: Change/Add Software Requirements
Procedure paper: Software Development Plan Development and Maintenance
Procedure paper: Software Acceptance Plan Development and Maintenance
Procedure paper: Software Test Report Development
Procedure paper: Software Test Plan Development and Maintenance
Procedure paper: Software Test Description and Maintenance
Procedure paper: Interface Design Document Development and Maintenance
Procedure paper: Version Description Document Development and Maintenance
Procedure paper: Software Design Document Development and Maintenance
Procedure paper: Interface Requirements Specification Development and Maintenance
Procedure paper: Software Requirements Specification Development and Maintenance
Procedure paper: Software Design Document Development and Maintenance
KPA Spotlights: Level 2 (RM) (technical report)
RM Overview from SEI work with a customer (slides)
Software Process Framework excerpts for RM
Requirements Management Flow: Ability to perform
Requirements Management Flow: Activities
Requirements Management Flow: Changes
Requirements Management Flow: Measurement
Requirements Management Flow: Measurement flowchart annex
Requirements Management Flow: Verifying
Requirements Management Flow: Verifying/Project Leader Review
Requirements Management Flow: Verifying/SQA Review
Requirements Management Flow: Verifying/Senior Management Review
Project Requirements Status spreadsheet

Table 2: Artifacts in the Guidebooks/Guidance Category

We were concerned that artifacts might be difficult to use due to their variability, the variety of contributing organizations, and their lack of integration. Yet one of the comments we heard consistently from potential users was the importance of seeing examples that had worked in practice. After evaluating the artifacts and determining that they could be useful independent of each other, we concluded that there was merit in presenting them without revision (except to assure donor anonymity).

3.3.2 Prototype Design

In the first workshop, participants imagined the RM transition package as a "blue box" full of materials that might be presented on paper, or that might be delivered using a variety of other media. As we began to accumulate artifacts for the prototype, we thought that the materials could be assembled into loose-leaf binders with a table of contents and an index. However, as we collected a few of the first and larger artifacts it was clear that many examples were being supplied in electronic format and that there was no benefit in also presenting these on paper. Also, there was clearly benefit in working with "soft" copies in terms of storage, handling, user tailoring, and distribution. These factors suggested to us that the transition package artifacts could be best managed and delivered if packaged as a secure World Wide Web site.

We saw other value in this mode of delivery. We could control access; and this was important because of the prototype nature of the transition package. We wanted to know who our users were and be able to survey them before and after their use of the site. Equally important, Web site activity logs would allow us to track who actually used the site, and gather data showing how they accessed the various artifacts. A Web site also would reduce production and distribution costs, because there were no paper, printing, or mailing costs.

Users first accessed the site at a public "front door" page that showed general information about the project and about transition packages. This page attracted interest from people beyond those we recruited personally. Several reviewers and organizations became trial sites after finding the RM TxP public page in a general search of the Internet.

Getting past the front door required a password, which gave access to an introductory page that described the concept and structure of the site. From there, a user could select one of three ways to view the contents of the site:

- by steps in the eight-step technology transfer model (Table 3), for those interested in artifacts to support particular stages in the introduction of RM [Fowler 96]
- by the common features of the Software CMM RM KPA (Table 4), for those interested in Software CMM coverage
- by category (example, template, checklist, guidance), for those who wanted to browse through artifact names

Activity 1: Establish the Change Team
Activity 2: Describe Desired State
Activity 3: Baseline Current State
Activity 4: Analyze the Gap
Activity 5: Develop the Solution(s)
Activity 6: Trial the Solution(s)
Activity 7: Roll Out the Solution(s)
Activity 8: Analyze Lessons Learned

Table 3: The Eight Activities in the Technology Transfer Model

Commitment 1: There is a written organizational policy for managing requirements
Ability 1: Responsibility is established for allocating requirements
Ability 2: Allocated requirements are documented
Ability 3: Adequate resources and funding are provided to manage requirements
Ability 4: RM training is provided
Activity 1: Allocated requirements are reviewed
Activity 2: Allocated requirements are used as the basis for software plans, work products, and activities
Activity 3: Changes to allocated requirements are reviewed and incorporated
Measurement 1: Measurements are used to determine the status of the activities for requirements management
Verification 1: Activities are reviewed with senior management
Verification 2: Activities are reviewed with project management
Verification 3: SQA reviews or audits activities and work products for managing requirements and reports results

Table 4: Key Practices for the Requirements Management KPA

These views of the information in the transition package served as indexes to artifacts. The views facilitated browsing and understanding of how the artifacts could be used in different contexts for RM implementation.

For most artifacts we prepared three formats:

- one for online viewing (in HTML)
- one that could be downloaded and used in least-common-denominator format (usually text)
- one that could be opened in its native application (e.g., Microsoft Project or Microsoft Excel) or opened in Adobe Acrobat for viewing or printing

Any of these versions of documents could be chosen by hyperlink from a menu page, which in turn was reached from an artifact link on one of the three view pages.

3.4 Evaluating the Prototype

The reason we built the prototype RM transition package was so that we could determine its value through actual use. In this section we describe

- who participated in prototype use
- the evaluation process
- pre- and post-trial surveys and responses
- our analysis of hits on pages in the prototype transition package Web site
- the implications of our findings from building and testing the prototype

3.4.1 Who Participated and How

We invited participants in the RM Workshop to evaluate the prototype as reviewers or as trial users. We also invited

- people at the SEI Symposium (August 1997) who stopped at a demonstration booth
- those who volunteered in response to presentations made to the Southern California SPIN and the Los Angeles SPIN meetings in September 1997
- those who found the front door to the RM TxP prototype Web site and inquired

For our prototype evaluation, we wanted as users people who were SEPG leaders or members of RM Process Action Teams, working in organizations that would be implementing RM during our trial period. We presumed that these would be the people who would benefit the most from the prototype TxP and would provide us with the most direct feedback on how useful it was and about how they used it. We also recruited, as reviewers, experts in RM, experts in technology adoption, and those who were interested but whose organizations were not at that time implementing RM and therefore didn't fit our user profile. Invited users and reviewers were screened; those who passed the screening were accepted as evaluation participants if they agreed to complete pre-use and/or post-use

surveys. Users completed both; reviewers completed only the post-use survey. All participants were guaranteed anonymity and were promised copies of any reports documenting the results of evaluating the prototype's use.

The Web site was available from late July 1997 through the end of October 1997. In February 1998, we conducted interviews with six of the twelve trial participants who participated as users to gain insight into how they had used the RM TxP since the end of the trial. Table 5 summarizes the contact and data-gathering efforts over the life of the trial project.

Stage	Number of people
• Pre-trial surveys requested and sent	92
• Pre-trial surveys returned	50
• Post-trial surveys returned (total)	25
– Participated as users	12
– Participated as reviewers	13
• 4 month follow-up with users	6

Table 5: Contacts with Trial Users and Reviewers

After the trial and review period was over, we compiled data from the surveys. We also examined over 14,000 hits on the Web site. Details on the pre- and post-trial survey data, and the Web site use data follow.

3.4.2 Pre-Trial Survey Results

Tables 6 and 7 show the pre-trial survey data, that reflect the variety of business and demographic characteristics of the 12 organizations participating as users.

Organization Size	Number of Organizations in Category	Number of Employees	Number of Employees Engaged in Software Development or Maintenance
Very large	2	4,000-40,000	800-3,500
Large	3	1,000 - 3,999	100-1,800
Medium	5	100 - 999	53-180
Small	2	Less than 100	24-40

Table 6: Range of Organizations Participating as Trial Sites, Part 1

Organization Size	Business Domains	Type of Software Developed or Maintained
Very large	Manufacturing; IT consulting for government	Information Systems (IS); telecommunications systems; embedded software
Large	Finance; systems integration; semi-conductor processing equipment; US government; manufacturing; transportation	Proprietary applications; telecommunications systems; IS; embedded software
Medium	Consulting; IS development; direct marketing; finance; medical equipment	IS; embedded software; user interfaces
Small	Aircraft; government contracting; consulting; government	Algorithm development; embedded software; IS

Table 7: Range of Organizations Participating as Trial Sites, Part 2

Note the range in size, business domain, and type of software represented in this group; transition packages seem to appeal to many organizations.

In the pre-trial survey we also asked for information about trial users' current RM practices:

- Were these practices defined, documented, and practiced as documented?
- How many requirements were managed in each release?
- How many requirements changed in each release?
- Were there measures in place for the current RM processes?

Most of the organizations participating in the trial did not have a defined and documented process for requirements management (recall that all of these organizations were seeking to move from Software CMM level 1 to level 2). Many did not count the number of requirements per release. Those that did also estimated the number of requirements that changed in each release. Organizations reported numbers of requirements per release ranging from 8 to 4000, and the percentage of requirements that changed by release from 17.5% to 50%; the size of the organization did not appear to relate to the number of total requirements per release or the number that changed per release. Clearly there were inconsistencies in how requirements were being counted. No organizations had measures in place for their RM process.

Finally, we asked about expectations of trial users for the RM TxP prototype, and what kinds of information they needed to better define and implement a more effective RM process. All the organizations participating in the trial said that they were in the midst of implementing requirements management and had an immediate need for assistance. They said they were looking for a set of materials to reduce risk in their implementation of RM. The benefit they saw in participating in the trial was to get early access to a product that could support that goal. None of the participants indicated interest in developing a transition package—as innovators and early adopters might have. Participants said that the materials in the package should be comprehensive, integrated, and easy to use.

3.4.3 Post-Trial Survey Response Analysis

After the trial, users and reviewers of the package answered post-trial survey questions. Their responses indicate that users spent considerably more time than reviewers with the prototype and intended to use the material from the prototype as part of their RM implementation. Reviewers spent less time with the prototype—an average of about an hour—and did not have a practical application to try it on. Thus, the analysis that follows is based on responses from the users, except where noted.

Among our survey respondents, the prototype was used most commonly by the leader of the RM introduction effort, or use was delegated by them to an internal or external software process improvement consultant working with the PAT or the SEPG. All users rated the quality of the TxP materials *useful* (3 on a scale of 1 - 4). Quantity of materials was rated by all between *just enough* and *too many* (2.2 on a scale of 1 - 3). All users said that they thought that the package should be delivered via a Web site.

With one exception, those working as change agents to introduce RM practice within trial organizations were new to RM and to the discipline of software engineering. For example, one participant spoke of his organization as having just lost a key person, the only one who really knew how to connect requirements to test. Despite the fact that all users were aware of the Software CMM and referred to the RM KPA as a standard or guideline, they told us they were seeking a better understanding of what it meant to practice requirements management. Apparently the CMM was not a substitute for tutorial information or detailed guidance. Although we considered the subjects of how to elicit and define requirements to be outside of the scope of the RM TxP, several users wanted more information on requirements definition and other requirements-related technical issues. They also wanted examples of "actual requirements" and asked for "contents of a software requirements specification" (SRS), even though these were beyond RM *per se*. The prototype did include one example of an SRS and one checklist for an SRS. It is not clear whether these were inadequate, not considered relevant to their problem, or were overlooked.

Several users downloaded all the materials for future use; not all users had time to explore all the materials. Materials that were used were most commonly adapted, or used for reference and comparison; some users used materials without modification. Those using the materials for reference said they were trying to get ideas about how to approach their RM problem; others wanted to compare their approaches to those of other organizations. One user said that having the collection of artifacts helped him and his group know which materials they needed to create: the collection itself acted as a checklist. Although users wanted examples from other organizations, they complained about things common to organization-specific materials, such as acronyms, different levels of detail, uneven quality, and inapplicability to small organizations as things that made artifact use more difficult. Their use of the pilot Web site appeared to be consistently ad hoc and opportunistic.

RM TxP prototype users provided many suggestions for its improvement. Not surprisingly, they wanted better integration of the technology transfer model with the artifacts, an area we knew was important but had not had the resources to address. One user stated that it was "too hard to find information." Others said they wanted

- "a road map"
- "better descriptions / indexing of materials"
- "a more practical transition model"
- "how to apply materials to a small organization"
- "implementation plans"
- "lessons learned"
- "varying examples by perspective"

These responses appear to imply that prototype users wanted a kind of primer for introducing RM, one that would spell out where to start and how to proceed, and one that was tailored to different organization types.

In addition, a few users felt the RM TxP prototype was missing some categories of materials such as

- "all the aspects of cultural change (guidance)—the really hard part" (non-RM specific)
- "incentives for behavioral change"
- "ability to 'auto-generate' the things I need"
- pointers to "other sources on level 2"
- "tool surveys and recommendations"

While some users of the site had access to similar materials from process asset libraries—or from the Web, consultants, books, or training—they all preferred to go to one source rather than several. Some users indicated that the RM TxP prototype saved them time and helped them to know what needed to be done. Two thirds of the users responding said they would purchase a product like the RM TxP prototype.[5] Nearly all organizations indicated they would contribute components to such a product.

3.4.4 Web Hit Analysis

Evaluating the actual use of the prototype by analyzing Web page hits yielded an objective view of what was accessed by the users and provides an interesting contrast to comments made in the post-trial survey.

Total number of accesses to the site's home page	100%
Subsequent hits for the view by artifact name	94%
Subsequent hits for the view by Technology Transfer Model (TXM)	68%
Subsequent hits for the view by CMM Common Feature	50%

Table 8: Page Access By View as a Percentage of Home Page Hits

Users of the prototype Web site, after they entered the site, generally used more than one view when they accessed artifacts (see Table 8). Users typically accessed multiple views in a session after accessing the home page. It is interesting that the view by Software CMM common feature was the least accessed view, despite the fact that in the post-trial survey respondents reported that it was the most popular view. One conclusion is that both the TXM and CMM views were useful, but not the primary mechanisms for browsing. The Artifact view supported browsing by name and the selection of general-purpose artifacts, which indicates to us that the users were looking for high-level and introductory information about requirements management and used the model-specific views opportunistically, for understanding, and to learn about the subject. This theory is supported by analysis of the survey responses that reported strong motivation to learn more about both software engineering and requirements management.

[5] A separate question sent out immediately after the survey asked if transition packages should be developed to support those introducing new practices in other technical areas; four of the five respondents to this question suggested transition packages for all the level 2 KPAs of the Software CMM.

The most accessed artifacts were general in nature. Table 9 shows the seven artifacts that accounted for 20% of the page hits on downloaded pages. This general information included the artifacts that described what requirements management is and looks like, and how to get the process action team started, including work group plans, charters, policies, etc.

RM "as is" Report
Training requirements checklist (from SEI course)
RM Technical Working Group Charter
Multi-Level partial strawman RM process flow
RM Technical Working Group Tactical Action Plan
Gap Analysis of "as is" RM Process, compared to the CMM RM KPA, from an RM Technical Working Group
Example RM policy (organization level)

Table 9: Top 20% Most Accessed Downloadable Artifacts

The 14 least accessed artifacts appear in Table 10. These artifacts generally are more detailed, concern implementation questions that become important late in the process of introduction, or are less obviously related to RM.

Requirements Management Flow: Measurement
Data Item Description for Consolidated Software Requirements Document (DID-CSRD)
Requirements Management Technical Note
Project Requirements Form
Requirements Management Flow: Verifying
Procedure Paper: Software Test Description and Maintenance
Procedure Paper: Interface Design Document Development and Maintenance
Procedure Paper: Version Description Document Development and Maintenance
Requirements Management Flow: Changes
Requirements Management Flow: Verifying/Project Leader Review
Procedure Paper: Software Design Document Development and Maintenance
Requirements Management Flow: Senior Management Review
Requirements Management "to be" process charts (in ETVX format)

Table 10: Least Accessed Artifacts (In Descending Order of Accesses)

3.4.5 Post Trial Follow Up

The questions we asked of the trial users four months after the end of the trial were designed to probe whether the trial organizations had made progress in RM-related process improvement, and whether their use of the RM TxP prototype contributed to that progress.

Of the twelve organizations that completed the post-trial questionnaires, we could reach only six for this longer-term follow up. For the six users that we could not contact, two people had changed jobs and no one had assumed their responsibilities, two were out of town during our survey period, and two promised feedback but were unable to provide it for various reasons.

Of those six users we reached, only two were still actively moving toward implementing requirements management. The other four had ended their efforts for a variety of reasons, including change in management sponsorship, reorganizations, deferral and absorption of the SPI effort into other organizational redesign efforts, and postponement of the effort due to workload. The two organizations that were actively moving forward used resources other than the prototype RM TxP, after evaluating it. For them, the lack of integrated case studies and artifacts in the TxP, the narrowness of its scope (one Software CMM KPA only), and its do-it-yourself nature limited its usefulness. Both organizations found support from consultants who provided integrated suites of materials and related support for implementation.

The other four organizations used the RM TxP primarily as an educational mechanism for the primary contact person and for others in their organization.

In all of these follow-up cases, the users of the RM TxP said that they gleaned ideas from it and that it contained enough artifacts. However, they proposed that it should contain more case study-like artifacts in different categories that fit together. Also, they asked for more than one example of the same artifact type, with examples differing by consistent criteria such as organization size, business domain, and application type.

We can't characterize the high drop-out rate among our users as being normal or unusual: broad studies of process improvement success rates in Software CMM level 1 organizations have not been published. Ninety percent of organizations that have reported an initial Software CMM assessment since 1991 have not reported a subsequent assessment. There are many possible reasons for this, including these: organizations conducted assessments but didn't report them; they used other methods for tracking their process improvement programs; their process improvement programs did not continue, etc. The parallels between our experience and these assessment data are provocative, but there is not enough information to draw conclusions.

3.5 Summary of Findings From Building and Testing the Prototype

In Appendix A we have described how to build transition packages, based on our experience in this project.

Users liked the idea of a transition package for RM, and, in general, they liked the content of the prototype RM TxP. Their responses confirmed the value of the many examples and templates for change teams and also for those performing RM processes. Finally, they said that they felt that use of the prototype saved them time and provided structure that they needed.

Almost all the organizations participating as trial sites had very limited internal resources for supporting their effort to introduce RM. They were looking for external sources for information and ideas. Thus, they appear to represent the majority adopter population categories for RM and the other level 2 KPAs of the Software CMM. They were looking for standard products and services to provide prepackaged solutions for use "as is" or with minimal adaptation. We believe these majority adopters can make good use of a transition package product that goes further than the prototype:

- that spells out how to orchestrate an RM change
- that provides materials and guidance for customizing the package's contents
- that is consistent with good software engineering practice

Despite the positive response to the prototype, the success rate of those in our sample attempting to implement RM was very low. We do not have a sample size large enough to claim that those who made progress toward implementing RM derived benefit from the prototype RM TxP. In the next section, we look at the results of the evaluation of the prototype from a broader perspective.

4 The Foundations of Transition Packages: Theory and Experience

Our research model—constructing a prototype transition package and then having organizations use and evaluate it—focused on proving or disproving the utility of the concept. The prototype was based on requirements from the November 1996 workshop and also on a set of assumptions we made, based on both research and practical experience in introducing software technology. In analyzing the data from the use of the prototype, we found it useful to state our assumptions explicitly and to examine these assumptions as part of organizing our findings. The following series of tables describe the assumptions in the categories of

- user characteristics
- in-use considerations
- content
- business concerns

Following each table we describe what we found that supports or disproves our assumptions, and where there are still questions.

4.1 Assumptions: User Characteristics

Table 11 lists the assumptions we made about user characteristics.

TxP users do not want to pioneer implementation of RM, they just want to use it.
TxP users have had limited exposure to and understanding of RM and the Software CMM.
Users of TxPs are SEPG members or PAT members.
Users know how to adapt artifacts for their own purposes.

Table 11: Assumptions About User Characteristics

4.1.1 TxP Users Do Not Want to Pioneer Implementation of RM

According to Rogers and Moore, "mass market" adopters—those from early and late majority adopter populations—of technology are not interested in the technology *per se* but in its application to their usual line of work [Rogers 95, Moore 91]. These adopters are focused on their business or occupation: banking, retail, aircraft, services, etc. They invest in acquiring technical know-how outside of their own business area only when they must.

Our users' comments, their recommendations for additional, more integrated materials, and the use of consultants by those who followed through and implemented RM, support the conclusion that they wanted the RM TxP to be completely turn-key, and to work with little tailoring or customization on their part.

4.1.2 Requirements Management and Software CMM Knowledge of TxP Users

The assumption that people who were candidates to use transition packages had had limited exposure to and understanding of the discipline of software engineering and the Software CMM, and are new to RM, is supported by both theory and experience.

In a study of how organizations adopt complex IT innovations (such as object-oriented design methods) Fichman and Kemerer claim that organizations are less likely to adopt these when "prior existing knowledge" is very limited [Fichman 97]. This raises the question of to what extent transition packages can compensate for this lack of prior existing knowledge, if at all. Most of the users reporting to us after the trial, and all of the users reporting to us in the 4-month follow up interviews indicated that they used the transition package, in essence, for educating themselves and their constituencies. Thus, one would assume they had little prior knowledge of RM. As a result of this feedback, we recommend that transition package developers take this early learning process into account. Conner and Patterson's work supports this suggestion if one assumes this learning process is similar to the "contact, awareness, and understanding" stages of their model of how people commit to change [Conner 82].

In addition to this apparent requirement for basic education and information on RM, there also appears to be a need for direct assistance. The two of our 4-month follow up participants who were making progress toward implementing RM had hired consultants. This seems to support a requirement for consulting-type help in adoption, when an organization is faced with the need to make real changes. Also, these users reported using the TxP for learning about RM before and while working with their consultants.

Consultants, thus, may play a necessary role in supplying the elements of a whole product when a technology is trying to break into majority adopter population acceptance, and whole products that include transition packages, training, and other components are not yet available.

4.1.3 SEPG and TWG Membership of TxP Users

We assumed that people who were likely to use TxPs are SEPG members or PAT members. People *pulling* technology into an organization assume the role of change agent. Change agents working in either software process improvement or technology change management are often organized in teams such as SEPGs and PATs. Rogers and others writing about change agents make it clear that one primary role of a change agent is translating the general version of a packaged product or technology for the specific context of their home organization. Thus, we assumed that transition packages could expedite that translation process by providing materials from which the change agent could learn and that they could adapt to their own situation.

The few people who were single-member SEPGs and PATs reported the least (or no) progress in the adoption of RM in our long-term follow-up interviews. Those who were more successful reported that they were tasked to implement process improvement, and were using the Software CMM and adapting the KPAs (RM among them) to their organizations. This is an example of change agent translation activity, whether they primarily used the trial RM TxP or other sources of information and examples.

4.1.4 User Ability to Adapt Artifacts

When we built the prototype RM transition package, we assumed that its users would know how to adapt artifacts—that is, they would act like early adopters of transition packages. Rogers and Moore provide evidence that early adopters prefer to use technology at a stage in its maturity when it is still malleable and incomplete enough for them to add to it. Early adopters have the interest and capability to make modifications to less mature technology, as described in Leonard-Barton's article of user participation in the design of an expert system at Digital Equipment Corporation [Leonard-Barton 87]. However, our users did not act like early adopters.

The feedback from all trial participants was that they would have liked to have seen artifacts that came from and were used by organizations like their own and that required little translation. While lack of these types of artifacts in the RM TxP prototype did not stop users from using artifacts they obtained from it, further feedback was that the most useful artifacts were those that were fairly general—whatever their source—and those that did not require adaptation.

As noted above, those users who were most successful in implementing RM got consultants to help them tailor, adapt, and develop the materials they needed rather than building the process support materials for themselves.

4.2 Assumptions: In-Use Considerations

Table 12 lists the assumptions we made about how people would use TxPs.

People undertaking a technology introduction effort want examples to use from organizations that have succeeded or that have had failures where lessons learned are available.
To get mass-market adoption, TxPs are only part of the whole product that should be supplied to the majority adopter population.
The Web is the best way to deliver TxPs for accessibility; the Web is the best way to support the hyperlinked nature of a TxP. (Artifacts need to link to each other and to frameworks, to facilitate selection of artifacts to download.)
Organizations implement a technology faster with TxPs than without.

Table 12: In-Use Consideration Assumptions

4.2.1 User Preference for Real-World Examples

The assumption that people undertake a technology implementation effort more readily if they have examples to use from other organizations that have succeeded or that have documented lessons from failures was strongly confirmed by those evaluating the prototype.

All of the trial participants who proceeded to implement RM capability used artifacts obtained from the TxP as well as from consultants and from training. Those who did not progress, but who had qualified to participate in the trial, also wanted access to the "real world" artifacts in the TxP to assist in their efforts. No participants attributed their lack of progress to a lack of access to example artifacts.

In the post-trial interviews and the four-month follow up interviews, most participants said that they wanted the artifacts to be characterized in some way, perhaps as parts of a series of case studies. We interpret this feedback as reflecting a need for understanding the context for different examples of the same kind of artifact.

Bloom and colleagues, in the well-known study that characterized a hierarchy of how people learn, indicated that people need examples early in their learning of a new subject area [Bloom 56]. Without examples, they cannot derive a cognitive map of the new area; and without the map, they cannot make analogies to their existing knowledge, a prerequisite for retention and application. Bloom does not, unfortunately, state whether one type of example is preferable to another. Other experts in instructional design such as Gagné might argue that well-constructed examples are superior to "found" examples (from experience), because

synthetic examples can be matched precisely to learning objectives [Gagné 85, Gagné 88]. The prototype users' desire for realistic examples from different domains may reflect a wish for examples that do not require extensive translation to be useful, along with a (perhaps) less reasonable belief that examples from outside one's own domain do not apply (the "not invented here" syndrome). If this assumption is correct future transition packages might be most effective using examples from experience as a basis for constructed examples more attuned to user domains.

4.2.2 Need for Whole Products

The theory behind the assumption that mass-market adoption success depends on supplying whole products to change agents in the majority adopter population is supported by the work of Moore and others. Moore says that "The single most important difference between early [adopters] and mainstream [adopters] is that the former are willing to take responsibility for piecing together the whole products (in return for getting a jump on the competition), whereas the latter are not" [Moore 91]. The phrase *technology transition* (from which we derived the name *transition package*) implies movement of technology from one place to another. It also implies that technology is actually used in everyday work once it is transitioned. Yet, when organizations acquire and try to apply new technology, they often act out of a simplistic view of what it takes to move technology into routine use.[6] Tornatzky and Fleischer state that "...inherently complex technologies . . . where significant amounts of organizational and social change are likely to characterize implementation" seldom are introduced and implemented without significant effort [Tornatzky 90]. Software and software-intensive technologies such as IT are arguably this complex, and thus require whole products to ensure successful introduction. What is not clear from this research literature is how much of a whole product is required. Those in the trial who made progress implementing RM supplemented the prototype TxP with training and consulting support, counter to our initial assumptions about how the TxP would be used. Our prototype transition package, containing examples, templates, checklists and written guidance, apparently did not reduce the need for other parts of a whole product such as training and the expert adaptation that consultants can provide.

4.2.3 Web Delivery Mechanism

The World Wide Web supports the hyper-linked nature of the TxP where artifacts need to link to each other and to frameworks in an easily browsed, easy to remember way; this organization also permits easy selection of artifacts to download. The medium encourages and invites participation. Thus, in this case, the need for theory may be obviated by overwhelming evidence from our prototype users and from experience with the Web as a whole.

---

[6] An earlier version of this discussion appeared in "Technology Transfer as Collaboration: The Receptor Group" [Fowler 90].

It was clear to us that there were logistical advantages to be gained from delivering the RM TxP prototype as a Web site. The only Web site-related complaints of consequence were that the prototype design wasn't easy to browse, that navigation to artifacts required extra mouse clicks, and that having to download one artifact at a time was inconvenient (many users wanted to download all artifacts and one person suggested a *zip* file to download everything at once). Regardless of these limitations of the prototype, Web-based design allowed for relatively easy implementation of multiple views within the TxP, and relatively quick and easy access to TxP artifacts by users.

Design issues aside, the ability to use the three views not just as indexes, but as the access mechanisms into the materials in the TxP, made the TxP concept easy to understand. It also made the prototype easy to maneuver, as demonstrated by the volume and pattern of "hits" on the site while it was active. Thus, a redesigned TxP, taking advantage of this type of hyperlinked structure, has the potential to encourage people to approach technology introduction work and problem solving in a variety of ways.

4.2.4 Faster Technology Implementation

That implementation of technology would occur faster with TxPs than without was an assumption that we did not address explicitly due to the short duration of the evaluation period. We did assume that people would spend less time finding an approach and materials for implementing RM because they would have a clear framework, such as that described in the technology transfer model, for how to proceed and because they would not need to reinvent materials through trial and error. We also expected that users of the prototype TxP would make progress in their process improvement efforts as a consequence of TxP use.

Of the 12 final participants, after four months two were nearly ready to implement RM and the others had postponed or stopped their improvement efforts. In the absence of studies that characterize normal improvement success rates for level 1 organizations we cannot tell if our two more successful organizations were typical, or to what extent the TxP may have helped them. No users reported problems caused by using the TxP, and all 12 of the users who finished the pilot reported that they used the materials in some way. This may have helped them move more quickly—we do not know.

4.3 Assumptions: Content

Table 13 lists the assumptions we made about the content of the TxP.

Examples are essential; examples from the real world are better. A mix of real-world examples is better than a carefully designed set of created examples from experts.
Examples need to be organized in a way that helps people find them. Tailoring guidance should be clear and explicit.
More examples (multiple individual artifacts, more different artifacts, more categories) are better than fewer.
There should be a rationale for why any example is included.

Table 13: Assumptions About Content

4.3.1 Example Types

The assumption that examples are essential and examples from the real world are better is discussed above in "4.2.1 User Preference for Real-World Examples." Nearly all of the people who responded to our questionnaires liked the fact that the artifacts came from real organizations; however, nearly everyone also wished that the context for each artifact had been described. For example, although the US Department of Defense (DoD) "flavor" of many of the artifacts was said to be a drawback by many of our users, the non-DoD users judged that most of the DoD-contributed artifacts were applicable to their needs. Many trial participants said that artifacts organized in case study form would have been more useful to them. Based on this feedback, we believe that TxPs should provide a view by contributor, because users could then see which artifacts had worked together in an organization. The risk with such a design is that some TxP users might overlook artifacts useful to them, simply because the characteristics of the contributing organization appeared to be incompatible. More investigation is needed in this area.

One of the more surprising lessons from the trial effort was that many of our users wanted access to the RM TxP prototype to learn about requirements elicitation and definition rather than about requirements management. There are a number of resources that describe how to develop and document requirements; for example Alan Davis' popular book on requirements engineering [Davis 93] and Wieringa's compendium of requirements specification frameworks and methods [Wieringa 96]. However, there are few collections of example materials available to the public. Therefore, it seems likely that the pilot participants are representative of a large community of individuals that would benefit from access to collections of materials, whether these provide basic information on the technology for awareness and understanding or give guidance and support about how to implement and manage a technology such as RM. The TxP seems to fill a desire for hands-on, interactive, practical, specific learning materials.

4.3.2 Organization of Examples

The assumption that examples needed to be organized in a way that helped people find them easily was readily born out in both research and experience. A mix of views into the set of artifacts (by Software CMM, steps in the Technology Transition Model, and type of artifact) proved to be a popular feature of the Web site. The TxP prototype users suggested two additional views: by contributor, and by adoption role.

Tailoring issues are more complex and difficult to sort out. Should there be guidance in a TxP on tailoring, combined perhaps with templates, or should a newspaper-like montage of artifacts be provided, with the assumption that users can do their own adaptation? We did not provide tailoring guidance in the prototype TxP for lack of donated tailoring examples; an early and ambitious plan to develop tailoring guidelines proved infeasible for this prototype.

4.3.3 Number of Examples

We assumed that more examples—multiple individual artifacts, more types of artifacts, and more categories—were better than fewer examples. When people see more than one example, they begin to compare and contrast them, and can determine the general principles that underlie the examples.

In any case, our trial users did not comment in any of the feedback that we had too few artifacts, although they suggested other artifacts they would have liked to have had; a few complained that there were too many. We did not attempt to establish where the thresholds were. However, from the beginning we sought donations of more artifacts, rather than fewer. It is relatively easy to find limited collections of integrated artifacts in the literature, or artifacts that can be acquired through training or developed with a consultant. It is harder to find alternative versions of the same artifact, or a set of artifacts drawn from experience that describe implementing and applying the technology in practice. Our trial feedback confirmed that this emphasis on quantity and variety is helpful.

4.3.4 Providing Rationale for Examples

Connecting each artifact to each view provided an implicit and rich rationale for why artifacts are included. Although our artifact links were not fully redundant and we did not provide descriptions of context or justification for including any of the artifacts, few of our users indicated concern about why certain artifacts were included or felt that others should not have been included. Most comments indicated that users understood why artifacts were in the TxP.

4.4 Assumptions: Business Concerns

Table 14 lists the assumptions we made about business concerns surrounding TxPs.

Many people will want to buy TxPs: This is an individual, not an organizational decision. To support this demand by individuals, TxPs should be inexpensive.
People prefer to get TxPs from the SEI rather than from other sources; people will contribute artifacts, especially if the TxP is sponsored by the SEI.

Table 14: Assumptions About Business Concerns

4.4.1 Price of TxPs

The assumption that acquiring products like the TxP is an individual, not organizational decision, and that TxPs should be inexpensive is supported in theory and in practice.

Moore claims that majority adopter populations want technology that has been transformed into an easy to implement and use commodity that is inexpensive [Moore 91]. In fact, the need to understand how to bring easy and systematic approaches to technology introduction for maturing technologies such as RM was the main reason for building the RM TxP prototype.

To reach the majority category of the adopter audience (who are price sensitive, yet will pay what is necessary to implement a standard), it seems important to keep the cost low. To confirm this assumption we asked our trial participants what they would be willing to pay for a production quality TxP.[7] The amounts they named were in the hundreds of dollars or less.

As with any product, the price of a TxP is constrained by the cost of development. For a Web-based TxP, it appears that cost is determined primarily by the strategy used to create content. If the content is gathered from contributors, there are costs for integrating and editing the artifacts for anonymity. If the artifacts are developed for the package from scratch, content-development costs may be quite high.

4.4.2 SEI Sponsorship

The assumption that people would prefer to get TxPs from the SEI rather than from other sources, and that people would contribute artifacts, especially if TxPs were to be developed by the SEI, was validated to some extent. When we asked for contributions to the prototype, people were generally eager to share their materials. This was also true of the participants in the evaluation when asked if they would contribute materials to future transition packages. Change agents may be predisposed to contribute because of the organizational recognition

[7] The cost to participate in our trial project was the cost of time for completing surveys and doing the interviews plus the effort to use the package.

and credibility they would gain, having materials accepted and "endorsed" by the SEI (if not in fact, by implication).

All of the participants wanted more SEI-sponsored materials to help them implement requirements management. The participants who succeeded with RM got help from non-SEI consultants and training. Despite this, they said they wanted to get materials from and contribute to an SEI project to create additional transition packages. They said they were less likely to contribute to proprietary consultant-developed TxPs. Our trial participants indicated that they felt the SEI has the responsibility to support not only new technologies with transition packages, but also the technologies referenced in models such as the Software CMM.

5 Conclusions

Introducing new software technology into widespread use is a difficult, risky proposition. Issues that impede technology adoption include

- the tendency to stay with the status quo even when present technologies perform similar functions less effectively than new technologies
- difficulty in selecting among competing technologies that attack similar problems but in a variety of ways
- non-technology issues that have a higher priority than new technology with potential users (e.g., cost, business strategy)
- the difficulty of developing a new infrastructure to support the use of the new technology

Any organization that wishes to reach their internal "mass market" of majority adopters with a new technology must solve these problems, which may have little to do with the new technology itself.

The idea to test the usefulness of TxPs for encouraging technology adoption came from the community of software engineering change agents and change sponsors. These people have worked with the SEI for years in achieving the SEI mission to "improve the practice of software engineering..." After many of the innovators and early adopters had installed and were using, for example, new processes and technologies that support practices described in the Software CMM, they had difficulty getting the rest of their organization (or their suppliers, or their customers) to implement the technology. In practice, it took additional support, tools, examples, and tailoring to expand the use of the practices and technologies after the initial successes.

We now realize that in adding support, training, tools, examples, and tailoring, these change agents were building whole products; that is what early and late majority users demand. In fact, in some of the organizations where commitment to process improvement has resulted in achieving the benefits of more mature processes, creation of transition packages as part of the process improvement process has become routine. Change agents in these organizations understand that what their innovators and early adopters create to implement a new process can be packaged and offered to later adopters, together with materials and experiences from

outside the organization. This reduces trial and error, and improves the confidence of later adopters that they will succeed at and benefit from the change.[8]

In this project we have moved from believing in the potential of TxPs to understanding how to build them. Based on the results of our RM prototype, we believe most technologies (except perhaps the simplest), as used by most adopters, will require other elements (for example, consulting support and training) to make the technology "go." We also believe that a transition package is a *major* component of the whole product for new technologies and processes. Regardless of the extent of competent consulting and good training, checklists, templates, written guidance and real-world examples are necessary. We believe providing these expedites the important learning that is prerequisite to making any technology or process-based change.

Any technology that is worth developing warrants careful attention as to how it will be implemented in organizations. For those technologies to reach a majority of adopters, transition packages are a necessary part of introduction planning and execution. Our prototype users confirmed that even our crude prototype-based efforts were helpful to them. TxPs built using the lessons from this project

- can be developed quite inexpensively
- can satisfy the expectations of the majority adopter population as they attempt to use new technology
- can speed adoption of the technology and will help its users achieve their goals

[8] Personal communication with Jock Rader, Craig Hollenbach, Brian Middlecoat and others, 1996-1998.

References

[Ackerman 83] Ackerman, A. F.; Fowler, P.J.; & Ebenau, R. G. "Software Inspections and the Industrial Production of Software," 13-40. *Proceedings of the Symposium on Software Validation*, Darmstadt, FRG, September 25-30, 1983. Amsterdam, The Netherlands: Elsevier Science Publishers B.V., 1983.

[Bloom 56] Bloom, Benjamin S., ed. *Taxonomy of Educational Objectives Book 1: Cognitive Domain*. New York: Longman, Inc., 1956.

[Conner 82] Conner, D. R. & Patterson, R. W. "Building Commitment to Organizational Change." *Training and Development Journal* (April 1982): 18-26.

[Culver-Lozo 95] Culver-Lozo, K. "Software Process Iteration on Large Projects: Challenges, Strategies and Experiences." *Software Process* 1, 1 (1995): 35-45.

[Davis 93] Davis, A. M. *Software Requirements: Objects, Functions, States*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[Dymond 95] Dymond, K.M. *A Guide to the CMM*. Annapolis, MD: Process, Inc. U.S., 1995.

[Fagan 76] Fagan, Michael E. "Design and Code Inspections to Reduce Errors in Program Development." *IBM Systems Journal* 3 (1976): 182-211.

[Fichman 92] Fichman, Robert G. "Information Technology Diffusion: A Review of Empirical Research," 195-206. *Proceedings of the 13th International Conference on Information Systems*. Dallas, Texas, December 13-16, 1992. New York, NY: ACM Press, 1992.

[Fichman 97] Fichman, Robert G. & Kemerer, Chris F. "The Assimilation of Software Process Innovations: An Organizational Learning Perspective." *Management Science* 43, 10 (October 1997): 1345-1363.

[Fowler 90] Fowler, Priscilla J. "Technology Transfer as Collaboration: The Receptor Group," 332-333. *Proceedings, 12th International Conference on Software Engineering*. Nice, France, March 26-30, 1990. Los Alamitos, CA: IEEE Computer Society Press, 1990.

[Fowler 94] Fowler, P. & Levine, L. "From Theory to Practice: Technology Transition at the Software Engineering Institute," 483-497. *Proceedings of the Hawaii International Conference on System Sciences*, Maui, HI. January 4-7, 1994. Los Alamitos, CA: IEEE Computer Society Press, 1994.

[Fowler 96] Fowler, P. & Kasunic, M. "Introducing KPAs: A Process Definition and Guide," Track 2. *Proceedings, Eighth Annual Software Technology Conference*. Salt Lake City, UT, April 1996. Ogden, UT: Software Technology Support Center, Hill Air Force Base, 1996.

[Fowler 97a] Fowler, P. & Patrick, M. "Experience Using a Benchmarking Workshop to Explore One Route to Practical Technology Introduction," 304-318. *Facilitating Technology Transfer Through Partnership: Learning from Practice and Research, Proceedings IFIP Technical Committee 8 Working Group 8.6 International Working Conference on Diffusion, Adoption, and Implementation of Information Technology*, Ambleside, Cumbria, U.K. June 25-27, 1997. London: Chapman & Hall, 1997.

[Fowler 97b] Fowler, P. & Patrick, M. *Proceedings of the Introducing Requirements Management into Organizations Workshop: Requirements Management Transition Packages* (CMU/SEI-97-SR-001, ADA325553). Pittsburgh, PA, November 11-13, 1996. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1997.

[Fowler 98] Fowler, P.; Patrick, M.; Carleton, A.; & Merrin, B. "Transition Packages: An Experiment in Expediting the Introduction of Requirements Management," 138-145. *Proceedings of the International Conference On Requirements Engineering*. Colorado Springs, CO, April 6-10, 1998. Los Alamitos, CA: IEEE Computer Society Press, 1998.

[Gagné 1985] Gagné, Robert M. *The Conditions of Learning and Theory of Instruction*, 4th ed. New York, NY: Holt, Rinehart, and Winston, 1985.

[Gagné 1988] Gagné, Robert M.; Briggs, Leslie; & Wager, Walter. *Principles of Instructional Design*, 3rd ed. Fort Worth, TX: Holt, Rinehart, and Winston, 1988.

[Goldenson 95] Goldenson, Dennis R. & Herbsleb, James D. *After the Appraisal: A Systematic Survey of Process Improvement, Its Benefits, and Factors that Influence Success* (CMU/SEI-95-TR-09, ADA302225). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1997.

[Grady 87] Grady, R. B. & Caswell, D. L. *Software Metrics: Establishing a Company-Wide Program.* Englewood Cliffs, NJ: Prentice-Hall, 1987.

[Hayes 95] Hayes, W. & Zubrow, D. *Moving on Up: Data and Experience Doing CMM Based Process Improvement* (CMU/SEI-95-TR-008, ADA300121). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1997.

[Hollenbach 96] Hollenbach, C. & Frakes, W. "Software Process Reuse in an Industrial Setting," 22-30. *Proceedings of Fourth IEEE International Conference on Software Reuse.* Orlando, Florida, April 23-26, 1996. Los Alamitos, CA: IEEE Computer Society Press, 1996.

[Jacobson 92] Jacobson, Ivar; Christerson, Magnus; Jonsson, Patrick; and Overgaard, Gunnar. *Object -Oriented Software Engineering: A Use Case Design Approach.* Reading, MA: Addison Wesley, 1992.

[Klein 95] Klein, Katherine J. & Ralls, R. Scott. "The Organizational Dynamics of Computerized Technology Implementation: A Review of the Empirical Literature." In Lawless, Michael W., and Gomez-Mejia, Luis R., *Advances in Global High-Technology Management* 5 (1995): Part A: 31-79. Stamford, Connecticut: JAI Press Inc., 1995.

[Klein 96] Klein, Katherine J. & Sorra, Joann Speer. "The Challenge of Innovation Implementation." *Academy of Management Review* 21, 4 (October 1996): 1055-1080.

[Kwon 87] Kwon, T.H. & Zmud, R.W. "Unifying the Fragmented Models of Information Systems Implementation," 227-251. *Proceedings of Critical Issues in Information Systems Research,* Chichester, England. New York, NY: John Wiley & Sons, 1987.

[Leonard-Barton 87] Leonard-Barton, Dorothy. "The Case for Integrative Innovation: An Expert System at Digital." *Sloan Management Review* 29, 1 (Fall 1987): 7-19.

[Leonard-Barton 92] Leonard-Barton, Dorothy. "The Intraorganizational Environment: Point-to-Point Versus Diffusion." In Williams, Frederick & Gibson, David V. (eds.), *Technology Transfer: A Communication Perspective*. Newbury Park, California: Sage Publications, Inc., 1990.

[Levine 94] Levine, L., ed. *Proceedings of the First International Federation for Information Processing (IFIP) Technical Committee 8 (TC8) Working Conference on Diffusion, Transfer and Implementation of Information Technology*, Champion, PA, 11-13 October 1993. Amsterdam: North Holland, Elsevier Science Publishers, 1994.

[McAndrews 97] McAndrews, Don; Marchok-Ryan, Janice; & Fowler, Priscilla J. *A Turbo-Team Approach to Establishing a Software Test Process at Union Switch & Signal* (CMU/SEI-97-SR-002, ADA324230). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1997.

[McFeeley 96] McFeeley, B. *IDEAL: A User's Guide for Software Process Improvement* (CMU/SEI-96-HB-001, ADA305472). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996.

[Moore 91] Moore, G. *Crossing the Chasm: Marketing and Selling Technology Products to Mainstream Customers*. New York, NY: Harper Business, 1991.

[Morton 83] Morton, R., ed. *Proceedings of the IEEE Computer Society Workshop on Software Engineering Technology Transfer*. Miami Beach, FL, April 25-27, 1983. Los Alamitos, CA: IEEE Computer Society Press, 1983.

[Olson 94] Olson, T.; Reizer, N; & Over, J. *A Software Process Framework for the SEI Capability Maturity Model*. (CMU/SEI-94-HB-001, ADA285595.), Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1994.

[Orlikowski 93] Orlikowski, W.J. "CASE Tools as Organizational Change: Investigating Incremental and Radical Changes in Systems Development." *MIS Quarterly* 17, 3 (Sept. 1993): 309-340.

[Paulk 94] Paulk, M. C.; Weber, C.V.; Curtis, B.; Chrissis, M.B. et al. *The Capability Maturity Model: Guidelines for Improving the Software Process*. Reading, MA: Addison-Wesley, 1994.

[Przybylinski 87] Przybylinski, S. & Fowler, P., eds. *Transferring Software Engineering Tool Technology*, Santa Barbara, CA, November 15-16, 1997. Los Alamitos, CA: The IEEE Computer Society Press, 1987.

[Rader 96] Rader, J. & Rundlet, N. "Teaching the Elephant to Manage Requirements," Track 6. *Proceedings, Eighth Annual Software Technology Conference*. Salt Lake City, UT, April 24-26 1996. Ogden, UT: Software Technology Support Center, Hill Air Force Base, 1996.

[Rogers 95] Rogers, E. *Diffusion of Innovations*. 4th ed. New York, NY: The Free Press, 1995.

[Strauss94] Strauss, Susan & Ebenau, Robert G. *Software Inspection Process*. New York, NY: McGraw-Hill, 1994.

[Tornatzky 90] Tornatzky, Louis G. & Fleischer, Mitchell. *The Processes of Technological Innovation*. Lexington, MA: Lexington, Books, 1990.

[Wieringa 96] Wieringa, R. J. *Requirements Engineering: Frameworks for Understanding*. Chichester, England: John Wiley & Sons, 1996.

[Zelesnik 92] Zelesnik, Gregory. *Software Requirements Engineering* [videotape series with instructor guide]. Available by email: customer-relations@sei.cmu.edu Software Requirements Engineering video purchase from Software Engineering Institute Continuing Education Series (1992).

Appendix A: Building Transition Packages

In this section we describe the process of building transition packages, and the reasons to build them.

A.1 Tips for Building TxPs

The artifacts that the users of our transition package used most were educational and general in nature. Also, our users liked authentic artifacts that had been used by successful teams. The advice here is very simple: beg and borrow examples of things that have worked in use. Look first for the overview, introductory materials used to introduce, manage, or operate the new technology. Give a lower priority to artifacts that describe the details of how it is engineered.

The process for building transition packages consists of the following seven steps:

1. Document a description of both the subject area for the transition package and the people you expect to use it. This description establishes the scope and purpose of your transition package effort.

2. Identify potential sources of materials.

3. Gather the materials.

4. Identify multiple views of the materials; if possible, base views on accepted reference models.

5. Assemble and package the materials, and create the views.

6. Distribute the package to the users.

7. Evaluate how people use the TxP and upgrade it accordingly.

A.1.1 Describe the Subject Area and Users

As with any product, the building of a transition package needs a clear set of requirements, and a clear statement of its scope. These requirements should be as precise as possible and should describe the technology that the TxP will introduce, the people who will use the TxP to introduce that technology (the change agents), and the people who will ultimately use the technology.

A.1.2 Identify Potential Sources of Materials

Example artifacts can be contributed by anyone, although innovators and early adopters are the most likely source. The identification of contributors involves research to determine who has succeeded in introducing the subject TxP technology into their organization, then contacting them and persuading them to donate their materials. Some means of locating the right people and organizations include: the Internet, conference proceedings, and referrals obtained through networking.

A.1.3 Gather the Materials

Ask for examples, templates, checklists, guidelines, tailoring notes, lessons learned, process descriptions. Any materials used in implementing or operating the technology may be useful.

Agreements with the donors of materials should specify

- how to modify the materials to disguise the contributor's identity
- terms of use
- restrictions on use

You may need to edit the materials to

- comply with terms of use agreements
- integrate the materials to some degree (be careful not to polish the materials too much)
- create a more "case study" appearance for the package, if that is particularly important to the users

Our RM TxP trial participants said that they would contribute to TxPs, especially if they received some benefit. They gave these examples of benefits: royalty on usage, price breaks on other products, and reciprocal access to artifacts. For the TxP pilot we found that organizations were willing to share their materials simply for being acknowledged as donors. In many cases, that may be sufficient compensation.

A.1.4 Identify Multiple Views of the Materials

In our trial of the RM TxP, the site statistics confirmed that most users found having all three views of the materials (by CMM, by technology transfer model, and by artifact type) useful. Providing different ways of accessing the materials is one direct way to support learning about the subject area. When users understand that particular artifacts are useful in multiple ways, depending on the frame of reference, comprehension of artifact application and use is accelerated.

We did not include a view in the TxP by user role. Comments from our users and examination of products similar to TxPs that were developed by others, suggest that we should recommend this way of organizing artifacts.

A common comment from trial participants was that they wished the materials had been more like a case study (explaining context and relationships among artifacts), or were characterized by type of contributing organization. Creating a case study from contributed materials is an expensive undertaking. A less expensive alternative is to create a view that groups artifacts by contributor (without necessarily naming the contributor), and that describes characteristics of the contributor and their application of the technology.

We recommend this minimum set of views for covering a set of artifacts:

- by type
- by reference model
- by introduction method
- by adopter role
- by contributing organization

Other views may also be useful; these should be developed and evaluated.

A.1.5 Assemble and Package the Materials and Create the Views

Hyperlinking TxP artifacts to views and possibly to each other means that distribution by the Internet or intranet makes sense. CD-ROM production and distribution could provide the same interconnected usefulness as Web-based distribution, but would slow access to, and complicate maintenance of artifacts.

For Web-based distribution, design considerations include these:

- the need to provide the artifacts in various formats (RTF, HTML, MSWord, etc.) so that the user can view, download, and use the artifacts opportunistically
- having the views positioned between the artifacts and the front page, so that the artifacts are *browsed* by way of the views
- making it as easy as possible to select and download the artifacts from the browsing page

A.1.6 Distribute the Package to the Users

Distribution of a Web-based product means providing users with access to the Web site, monitoring whether and how they are accessing the TxP, and providing the usual support required by online users.

Conditions can be placed on access; for example, supplying information describing intended use of the package, or describing goals for technology adoption may be a prerequisite for access to the site.

A.1.7 Evaluate and Upgrade the Transition Package

The tools that Web-based content delivery provide enable you to assess whether the transition package is being used, what pages are being used, and who is accessing the Web site. Regular review of these Web-server-provided statistics will show whether there are particular artifacts that are useful or not useful.

Ongoing contact with the users, to determine whether they are meeting their adoption goals and how the TxP is supporting them, can lead to improvements in the TxP.

A.2 Reasons to Build Transition Packages

Introducing a new technology is an undertaking that is, statistically, likely to fail [Moore 91]. Even a well-understood, superior technology can be ignored by potential adopters. Transition packages are a means of reducing the barriers to the introduction of a technology, particularly after that technology has been adopted by the innovators and early adopters and is in jeopardy of being rejected by the majority of the adopter populations. Those with responsibilities for developing and gaining widespread adoption of a technology can increase the odds for success by building transition packages.

In sum, here are reasons to build a transition package:

- TxPs codify the understanding of a technology.
- Multiple examples enable an understanding of the technology if based on multiple complementary perspectives.
- Easy access and the freedom to tailor and modify the examples encourages experimentation and learning.
- Barriers to introducing the technology caused by a lack of examples in the introduction of the technology are removed.
- A description of how to succeed in using the technology is presented.

These are things that are important to the early and late majority adopters, those users of a new technology who are adopting it because it represents a standard that they must meet to perform their primary business or because it has clear value to them with limited and specific risks. These adopters are not interested in pioneering processes to use a new technology for competitive advantage—they just want it to work. The collection of examples in a TxP supports rapid learning, and the materials can be useful "out of the box" for implementing the new technology. For a technology to break through into widespread adoption, a whole product is necessary. TxPs are a necessary part of the whole product for any technology.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (LEAVE BLANK)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	July 1998	Final

4. TITLE AND SUBTITLE
Transition Packages for Expediting Technology Adoption: The Prototype Requirements Management Transition Package

5. FUNDING NUMBERS
C — F19628-95-C-0003

6. AUTHOR(S)
Priscilla Fowler and Malcom (Mac) Patrick

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

8. PERFORMING ORGANIZATION REPORT NUMBER
CMU/SEI-98-TR-004

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
HQ ESC/DIB
5 Eglin Street
Hanscom AFB, MA 01731-2116

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ESC-TR-98-004

11. SUPPLEMENTARY NOTES

12.A DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified/Unlimited, DTIC, NTIS

12.B DISTRIBUTION CODE

13. ABSTRACT (MAXIMUM 200 WORDS)

This report describes the experience of building and evaluating a prototype transition package for organizations implementing processes in support of the Requirements Management Key Process Area of the Software Engineering Institute's Software Capability Maturity Model℠. This report also describes the theories that led to the project and states our conclusions based on evaluation and review of the prototype by users typical of the audience targeted for transition packages. Feedback from these users indicated that they were typical "early or late majority" adopters, and that they found the transition package helpful for orientation and education as part of implementing requirements management practices in their organizations. We argue in this report that transition packages, as part of a complete "whole product" that includes training and consulting, can be an effective mechanism for expediting the diffusion, adoption, and implementation of important technologies. Finally, we describe what we now know about creating transition packages and how they might be used.

14. SUBJECT TERMS: transition package; technology transition; process improvement; whole product; technology adoption; technology transition; diffusion of innovation; implementation; software engineering; requirements management; process action teams; change agents; software engineering process group

15. NUMBER OF PAGES
50

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL